IWANN'95 proceedings - International Workshop on Artificial Neural Networks, Malaga (Spain), 7-9 June 1995
From Natural to Artificial Neural Computation, J.Mira, F.Sandoval eds., Springer-Verlag, Lecture Notes in Computer Science 930, 1995, pp.696-703

A VLSI SYSTEM FOR NEURAL BAYESIAN AND LVQ CLASSIFICATION

Philippe Thissen*, Michel Verleysen*, Jean-Didier Legat*, Jordi Madrenas†, Jordi Domínguez†

* Université catholique de Louvain,
Laboratoire de Microélectronique - DICE,
3 Place du Levant, B-1348 Louvain-La-Neuve, Belgium
† Universitat Politècnica de Catalunya,
Departament d'Enginyeria Electrònica
Gran Capita, s/n, E-08071 Barcelona, Spain

Various types of neural networks may be used in multi-dimensional classification tasks; among them, Bayesian and LVQ algorithms are interesting respectively for their performances and their simplicity of operations. The large number of operations involved in such algorithms may however be incompatible with on-line applications or with the necessity of portable small-size systems. This paper describes a neural network classifier system based on a fully analog operative chip coupled with a digital control system. The chip implements sub-optimal Bayesian classifier and LVQ algorithms.

1 Introduction

Many neural network methods have been developed recently to solve the problem of multi-dimensional data classification; among these methods, we can select sub-optimal Bayesian classification (see for example [1] and [2]) for its performances and its convergence to the (optimal) Bayesian boundaries between classes, and LVQ (Learning Vector Quantization) [3] or 1-Nearest-Neighbor for its simplicity of operations.

Without going into the details of the equations, we can mention the principle of these two classes of algorithms. In the first one, the principle is to approximate the true probability density of learning vectors, by superposition of Gaussian (for example) kernels [4][5]; the idea is to place on each vector of the learning set a radial-basis kernel, which is usually chosen as Gaussian, and to sum all such kernels class by class; it may be proven that if the number of learning points is large, this approximation will converge to the true probability density of vectors in each class. Once the probability densities have been estimated, the Bayes law can be used to select the most probable class to attribute to a vector to classify, by choosing the class having the largest product between the estimate of probability density at the location of this vector, and the a priori probability of the class, this last one being estimated by the ratio between the number of learning points in each class and the total number of points. Such a Bayesian classifier however requires a large number of Gaussian functions to obtain acceptable approximations of probability densities. In order to reduce the number of computations, sub-optimal methods have been developed [1][2][6];

they mainly consist in selecting a reduced number of kernels, their locations being fixed by a vector quantization method, and their width by some function of the variance of each cluster.

The LVQ principle is even more simple: once the prototype vectors (equivalent to the centers of the radial functions in the Bayesian classifier) have been located by some adaptive method of vector quantization, the principle of the classification by LVQ [3] is to attribute to a vector to classify the class of the nearest prototype, using either an Euclidean either a Manhattan distance.

2 Architecture of the ANKC processor

Because of the high number of computations involved in sub-optimal Bayesian and in LVQ classifiers, it may be interesting in some applications to have a VLSI specialized chip implementing these two kinds of algorithms; the advantage of a specialized chip in this context is not only to offer a high speed of computation (see evaluations in section 6), but also to realize portable classification systems, without the need for cumbersome general-purpose computers.

The general architecture of the ANKC (Analog Neural Kernel Classifier) processor, designed to implement sub-optimal Bayesian and LVQ classifiers, is detailed in figure 1.

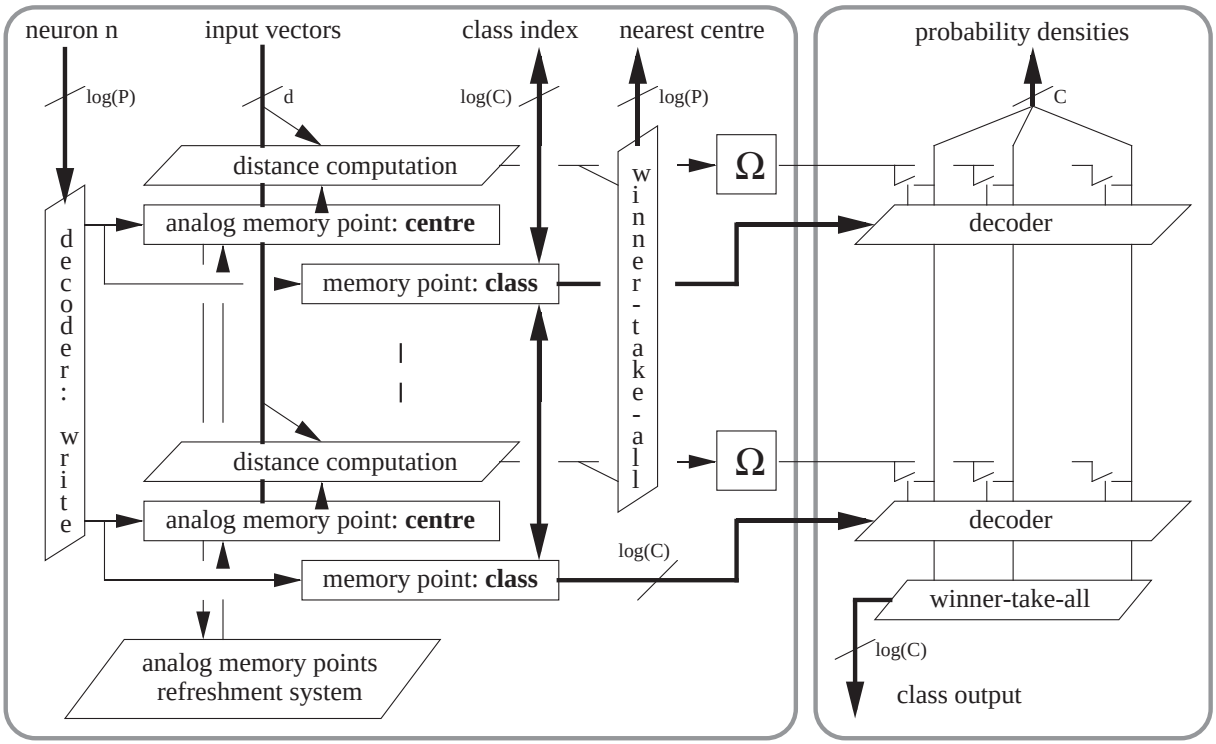

Figure 1: Functional description of the analog processor

The core of the system is built around P identical cells, each of them being composed of memory points to store the coordinates of the centroid (prototype vector) and its class, together with a distance calculator to compute the distance between this centroid and an input vector. Shortly, the system will work as follows. A set of P centroids will be stored in the processor; in the case of the Bayesian algorithm, the coordinates of the centroid correspond to the center of the kernel function Ω. Then, when an input vector is presented to the circuit for classification, all distances between this input vector and each of the centroids are computed in a parallel way; this is the purpose of the P mentioned distance computation cells.

The P computed distances are then used in two ways. On one hand, they are compared to find the smallest one, in order to select the closest centroid from the input vector; this is used in LVQ-like algorithms, in the purpose of selecting the winning centroid. On the other hand, the distances serve as inputs to P Gaussian-like kernel functions, used in Bayesian algorithms.

In the case of the LVQ algorithm, the selection of the winning centroid completes the recognition phase of the algorithm. In the case of the Bayesian algorithm, the P kernel outputs are summed class by class, in order to estimate the probability density of each class. The parameters of the kernels, namely their widths and shapes, may be adjusted by external commands. According to the Bayes law, classification of the input pattern is then realized by selecting the largest probability density among the different classes. In the Bayes law however, the probability density estimates must be multiplied by the a priori probabilities of the classes, before selecting the largest value; this is verified in our circuit if the number of centroids in each class is proportional to these a priori probabilities, which is the usual condition required on the centroids in all vector quantizations algorithms used for classification.

3 Test processor

A test version of the ANKC processor has been designed according to the above architecture. The chip has 16 centroids in dimension 8, each of them belonging to a maximum of 8 different classes.

The locations of the centroids are memorized in analog memory cells; all cells are detailed in reference [7]. The principle of the analog memory cells is to memorize a voltage on a capacitor by a cascode current copier cell. Leakage currents and charge injection may disturb the memorized value; the circuit has been designed in order to make the charge injection negligible in comparison with one LSB of the dynamics, and to make the leakage currents small enough so that a value is stable on the capacitors during approximately 1/10 sec., at the precision of one LSB. The accuracy of each memory point is 7 bits.

Each regulated cascode current copier used as analog memory point is connected to an input circuitry, also implemented as a regulated cascode cell. The principle of the two algorithms described above is to compute the distances between an input vector and each of the prototypes; to simplify the operations, the Manhattan distance is used in the system (the use of one type of distance or another only influences the performances of the classification in a negligible and non-systematic way). The input circuitry will thus produce a current in the same range as the current memorized in the cell, and their subtraction component by component and summation on the 8 components per vector by Kirchoff's law will produce the Manhattan distance between the input vector and one of the memorized ones. Before summation, positive and negative currents are separated in order to sum their absolute values.

The result of the Manhattan distances computation is fed either in a looser-take-all circuit for LVQ purposes (to select the smallest of these distances), either in a Gaussian kernel for the Bayesian classifier. The analog cells implementing the kernel functions have been designed to realize Gaussian-like functions with both adjustable slope and width, allowing thus two parameters that can be used to improve the performances of the classifier. Slopes and widths are identical for all kernel functions on a chip.

We will describe into more details the analog memory points refreshing system. This system converts the current memorized in a memory point by a successive approximation scheme (SAC) based on the propagation of a token in a finite-state automata [8]. In figure 2, at the first step of the SAC the memorized current is first compared to a current equal to 2^6 LSBs; the two (opposite) currents are serially connected at point C, and the voltage at this point rapidly goes high or low respectively if the memorized current is smaller or higher than the source. Voltage C is then compared with the medium point of the dynamics, and the result of the comparison determines if the source of 2^6 LSBs will remain connected during the next phases of the SAC or not (if C is higher than M, it means that the source current of 2^6 LSBs is smaller than the memorized current, and thus the source will remain connected during the next phases; if C is less than M, it means the source current is too high, and it will thus be disconnected for the next phases).

Latches in figure 2 will not only transfer the token at each step, but will also memorize the status of each successive comparison. Once the decision about the MSB has been taken, the

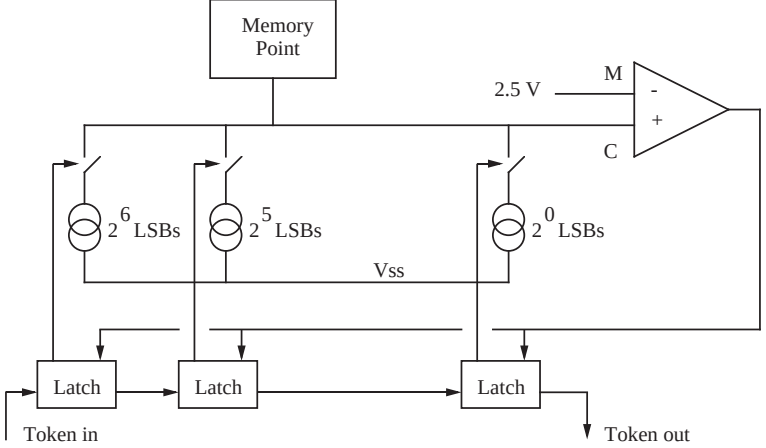

Figure 2: Principle of the refreshment system for analog memory points

token will be transmitted to the second cell, and the decision about the switching on or off the 2^5 LSBs source will be taken, and so on... After 7 iterations, the sum of all the sources of 2^i LSBs ($0 \leq i \leq 6$) which has been switched on during the approximation process will constitute a lower estimation of the memorized current.

The memory cell has been designed to ensure that the leakage currents will always have the same sign; in other words, it ensures that the current memorized in the cell will always decrease. Refreshing the memorized current by a value equal to the above approximation plus one LSB will thus ensure to keep the memorized current fixed at the precision of one LSB, if the leakage currents between two successive refreshments are kept under this limit. Simulations showed that a delay of 0.1 sec. between two successive refreshments may be chosen, for a capacitor value in the cascode memory point equal to 1 pF, and a precision of 7 bits in each cell, even with large overestimations of leakage currents and parasitic effects. This means that the whole circuit has to be switched in refreshing mode only each 0.1 sec.; since the refreshing time for the whole circuit (128 cells) will be around 1 msec. (with a clock frequency of the refreshment system of 1 MHz), this means that the percentage of time that the circuit has to be switched in refreshment mode is about 1%, which is more than acceptable; during the other 99% of the time, the circuit can be used in classification mode. The two modes cannot be mixed, since the refreshment modifies the voltage at node C, and the current flowing from a memory point will thus be corrupted; one mode or another must be chosen through a set of switches connected to an external control line.

Connection of the refreshment system to one memory point or another is assumed by two multiplexors connected to switches not represented in figure 2. The state of these multiplexors (line and column of the memory points array) is determined through two shift registers; the first one (row) changes its state each time the token in the refreshment system goes out, the second one (column) each time the row shift register achieves a complete cycle. By this way, all memory points will be successively connected to the successive approximation analog-to-digital convertor, and refreshed. The end of the cycle of the column shift register will signify the end of a complete refreshing cycle.

Let us finally mention that the use of the refreshment system is not the only possible way of refreshing the values stored on the capacitors; another way is to duplicate all memorized values in a digital external memory, and to periodically refresh the internal memory points with the values stored in the external ones. Because of the strongly reduced number of steps involved in a refreshment of the complete memory array, the global refreshment time is now reduced to approximately 0.02 sec., which is an obvious advantage of this solution, the drawback being the more complex peripheral components and control system to be used.

4 External control and setup

A setup for the prototype system is shown in figure 3. To keep the system portable, an embedded specific controller is used, but connection to a host could be done as well. The ANKC test chip is digitally controlled by means of an FPGA and an programmable read-only memory (E)PROM. The former provides the required control signals to the ANKC processor, and the latter contains the programming information of the analog processor. Two switches (that could be physically hardwired to reduce components) determine if the system performs a previous digital test, and the internal or external refreshing mode.

The functions performed by the external digital control of the ANKC processor are the following: clock generation, digital test, processor initialisation and analog memory point refreshing in external mode.

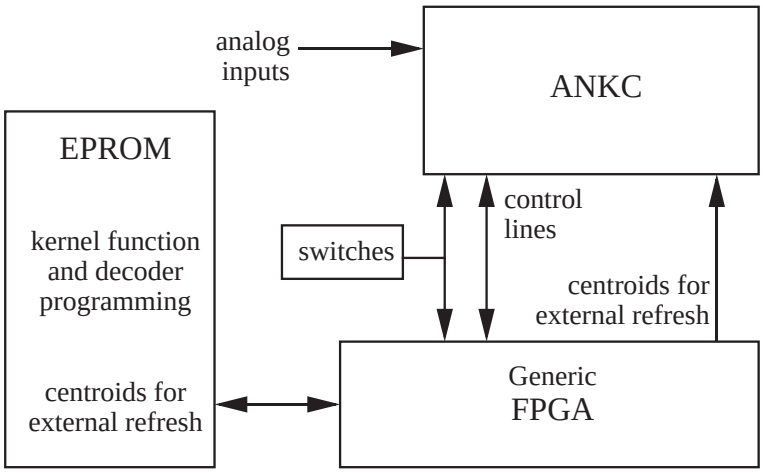

Figure 3: Control architecture of the ANKC system

- Digital test: The ANKC internal latches form in a test mode scan-path chains that allow to feed test vectors through a serial line and thus bring the digital part of the processor to a known state. The output stream of the scan-path chain is compared by the controller with the previously entered data in order to verify if any bit has been altered. In the prototype system only the test of the processor digital part is considered, but it can be easily extended to test the analog function, provided analog inputs and output measurement are available. The test is performed if the associated switch is active.

- Processor initialisation: After testing the digital part, the decoder switches and the width and slope of the kernel functions (figure 1) are programmed. Then, the row and column registers are reset and the analog memory points are programmed through the D/A converter input lines. All the information needed in this step is stored in the (E)PROM.

- Refreshing: During normal processing, as indicated in the previous section, classification and refreshing are interleaved. During refreshing, if internal mode, the controller just counts the refreshing interval and activates the refresh signal. In external refreshing mode, it also enters the digital value of the memory points.

5 Accuracy of computations

Determining the necessary accuracy for all computations in an analog chip is not obvious. Limitations on accuracy mainly come from charge injection, leakage currents, mismatching between devices and noise (like power supply noise). In the ANKC processor, charge injection and leakage

currents affect the values memorized on the capacitors in the analog memory points; we showed however in a previous section how this problem has been addressed, by periodic refreshment of the stored values. Mismatching between devices is also much more critical than possible noise; the following shows how mismatching has been modeled and how it affects classification results in a suboptimal Bayesian classifier (the IRVQ [2] algorithm was used for the simulations).

Mismatching between transistors may be modeled by a circuit simulator as HSPICE. However, in the case of a complex analog circuit as ANKC, it is out of question to try to simulate the whole chip by a circuit simulator; electrical simulations are only carried out at the cell level, and to verify the connections between cells. To test the influence of mismatchings on the classification task itself, a more reasonable way on a computational point-of-view is to model the parasitic effects in a simulation of the classification algorithm itself.

Three mains steps are achieved in the ANKC processor to classify a vector: Manhattan distance between the vector and the centroids, non-linear (Gaussian) functions of the distances, and winner-take-all to select the winning class. In the Manhattan distance computation, the crucial point resides in the current mirrors which have to be used to sum the currents and to take their absolute values; mismatching at this level has been modeled by adding a white noise at the output of the each distance computation block the amplitude of the noise being determined by a constant fraction of the total dynamics, expressed in bits (a precision of b bits corresponds to noise dynamics equal to $1/2^i$ of the total dynamics).

Concerning the Gaussian functions, simulations showed that the exact shape of the function has no influence on the classifications results. What is more important is to select the width and the slope factors of the curve; these parameters have been chosen in these simulations as those giving the best classification results by the IRVQ algorithm implemented on the chip without noise and mismatchings (ideal implementation of the algorithm).

Finally, the mismatching in the current mirrors used in the winner-take-all has also been modeled by adding a white noise value at the input of this device, as after the distance computation. We may consider that the influence of these two noise sources will be approximately equal to the influence of mismatchings in the distance computation and winner-take-all devices, with an identical number of bits defining the dynamics of the range and the percentage of mismatching in current mirrors.

Simulations have been made with the IRVQ [2] algorithm. They are illustrated here on the IRIS database (150 points in dimension 4, and 3 classes), which is a standard in classification studies; simulations were however carried out on a large number of databases (reals ones coming from the industrial world and artificial ones generated for special tests), and all qualitative results are similar.

Figure 4 illustrates the classification error on this database in function of the dynamics of the two sources of noise described above; we can see that the performances of an ideal circuit are very closely approximated if the precision of the mismatched devices is around 7 bits, while acceptable performances are already obtained for 3-4 bits accuracy; these results were used in the implementation of the ANKC processor, by taking the necessary precautions to design the critical current mirrors with an accuracy of about 7 bits.

6 Performances

Evaluating the performances of such analog processor greatly depends on the type of measure used for the evaluation; comparing serial digital and parallel analog processors in terms of FLOPs is not fair, because of the very different nature of operations between processors. In the following, the performance of a processor is thus measured in terms of number of operations that should be used on a serial digital processor to realize an identical set of computations (GOPS = 10^9 operations per second).

Giving this definition, we can evaluate the number of these operations that are computed in

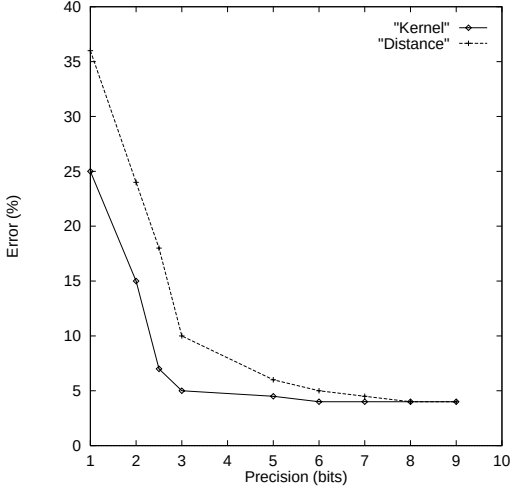

Figure 4: Influence of the two main sources of mismatching in the ANKC circuit on the classification error by the IRVQ algorithm on the IRIS database

one *clock cycle* (period between the presentation of two successive input vectors) in the ANKC processor. If N is the dimension of the vectors, P the number of centroids in the ANKC processor, and C the number of different classes, we have:

Task	**Number of operations**
P distance computations	$3.P.N$ additions
P kernel functions	$P.\log2(N)$ comparisons
P additions	P additions
1 winner-take-all over C values	C comparisons
1 winner-take-all over P values	P comparisons
Total	$3.P.N + P.\log2(N) + 2.P + C$

In our implementation of the ANKC test chip, we have $P = 16, N = 8$ and $C = 8$; with a clock of $10MHz$ (this frequency is used as the main clock of the circuit, so that one vector is classified at each cycle), this leads to a number of operations (as defined above) per second equal to 4.72 GOPS. The chip size in MIETEC 2.4 μm technology is 5x4 mm.

To compare the performances of the ANKC processor with another up-to-date digital processor implementing similar algorithms, the *Ni 1000 Recognition Accelerator* from INTEL, we have to take the technology and size of the circuits into account. The *Ni 1000* chip is fabricated in a $0.8\mu m$ technology, and occupies 12x12 mm. It has been evaluated that an ANKC processor with $P = 256, N = 32$ and $C = 16$ can be implemented in the same technology on the same silicon area. Evaluations of the performances according to the above definition then give 264 GOPS for the ANKC processor, and 16 GOPS for the *NI 1000*, which shows the strong advantages obtained with a fully analog implementation of such kind of classification algorithms.

7 Conclusion

The analog implementation of a neural network classifier, based on suboptimal Bayesian or LVQ algorithms, has been described in this paper. The analog operative chip is coupled to a digital control part, to form a complete, portable (small size), powerful classification system. It implements the classification phase of the algorithms, while learning has to be performed on an external computer and the results of the learning downloaded in the system. A test chip has been designed in MIETEC 2.4 μm technology, with a reduced number of cells. Evaluations show that the per-

formances of this classification system surpass those of an up-to-date digital chip implementing similar algorithms by a factor around 16.

8 Acknowledgments

Part of this work has been funded by the ESPRIT-BRA project 6891, ELENA-Nerves II, supported by the Commission of the European Communities (DG XIII). Philippe Thissen is working towards the Ph.D. degree in microelectronics under an IRSIA (Institut pour l'Encouragement de la Recherche Scientifique dans l'Industrie et l'Agriculture) fellowship. Michel Verleysen is a Senior Research Assistant of Belgian National Fund for Scientific Research (FNRS). Jordi Domínguez holds a Research fellowship (FPI) from the Spanish Ministry of Education and Science. We thank Olivier Herbeuval and Yves Van Daele for their contribution in the design of the ANKC test chip.

References

[1] P. Comon, G. Bienvenu, and T. Lefebvre, "Supervised design of optimal receivers", in *NATO Advanced Study Institute on Acoustic Signal Processing and Ocean Exploration*, Madeira, Portugal, July 26-Aug. 7 1992.

[2] J.L. Voz, M. Verleysen, P. Thissen, and J.D. Legat, "Handwritten digit recognition by suboptimal bayesian classifier", in *NeuroNîmes94 (Neural Networks and their applications)*. October 1994, EC2, submitted.

[3] T. Kohonen, *Self-Organization and Associative Memory*, Springer-Verlag, Berlin, 1989, 3rd Edition.

[4] T. Cacoullos, "Estimation of a multivariate density", *Annals of Inst. Stat. Math.*, vol. 18, pp. 178–189, 1966.

[5] E. Parzen, "On the estimation of a probability density function and the mode", *Ann. Math. Stat.*, vol. 27, pp. 1065–1076, 1962.

[6] Q. Xie, C. A. Laszlo, and R. K. Ward, "Vector quantization technique for nonparametric classifier design", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 12, pp. 1326–1330, december 1993.

[7] M. Verleysen, P. Thissen, J.L. Voz, and J. Madrenas, "An analog processor architecture for neural network classifier", *IEEE Micro*, vol. 14, no. 3, pp. 16–28, June 1994.

[8] D. Macq, J.D. Legat, and P. Jespers, "Analog storage of adjustable synaptic weights", in *Proceedings of the SPIE conference on Applications of Artificial Neural Networks, Orlando (USA)*, April 1992, pp. 712–718.